Paper 4673-2020

SAS® Event Stream Processing at the Edge: Reduce or Eliminate the Need to Transmit Data for Analysis

Bob Augustine, Hewlett Packard Enterprise

Sri Raghavan, Hewlett Packard Enterprise

ABSTRACT

The new frontier is the Intelligent Edge. The intersection of people, places, sensors, things and computers defines the intelligent edge.

HPE tested SAS ESP using NVIDIA Tesla T4 GPUs that were installed in an HPE EL4000, along with m510 server cartridges. This allows for video processing to be performed right at the edge, which reduces or eliminates the need to transmit large amounts of raw data to the data center or cloud for analysis. There will still be times when a subset of summarized data needs to be transmitted to the data center for further analysis. For example, immediate analysis can happen at the edge, but summarized data transmitted from the edge to the data center can then be combined with other like sources to look for trends across a larger sample.

These trends could be looking for issues where a design update may be required or they could be looking at how environmental factors affect equipment in the field. Also, we will show how visual data captured by drones that can be used to determine when maintenance is required, rather than performing maintenance on a time basis.

INTRODUCTION

Today, more data is being generated, captured, and analyzed than ever before. Data is being captured from devices across industries at the intersection of people, places, and things. The intelligent edge is when we have the infrastructure to act upon the data where it is generated, as it is generated.

The cost of transmitting data from the edge to the core or cloud is directly proportional to the amount of data that has to be transmitted within a specified, constrained time period. There was a time when top line local area network speeds were 10Mb or even 100Mb per second. Now we are seeing network speeds of 10Gb, 25Gb and even 100Gb per second. That's just for local traffic. What happens when we have edge systems located in service vehicles or in other remote locations that cannot be serviced with local area types of networking? Then we need to rely on cell technology, DSL, cable or other, slower transmission protocols to transfer data from the edge to the core or cloud, which may be more expensive on a per MB basis and slower. (Note: According to several sources, including Deutsche Telekom, 5G will potentially top out at about 10Gb/sec in the future[1], because it limits the amount of data that can be transmitted within a specific period of time.)

Transmission of data also introduces latency from the time the information is collected until it has accrued enough to begin the analysis of that data. Latency has the potential of reducing the efficacy of the analysis, or worse, may call the success of the analysis into question. In addition to the expense associated with data movement, when we transmit data from external sources, we introduce security concerns that must be addressed to

[1] Found at https://www.telekom.com/en/company/details/5g-speed-is-data-transmission-in-real-time-544498

ensure the data and the systems are not intruded upon. It is for this reason that we try to process as much data at the edge as possible and only transmit a subset of the data to the core.

In this paper we will look specifically at video analytics processing at the edge. We will demonstrate the differences in processing speeds attainable by installing an NVIDIA Tesla T4 GPU in an HPE Edgeline product such as the HPE Edgeline EL4000, and we will propose a scenario that will provide a return on investment. The solution that we're highlighting in this document is made up of SAS Event Stream Processing (ESP) and HPE Edgeline servers.

SAS ESP allows the user to perform advanced analytics while trapping streaming data as it comes from sensors and other edge devices.

HPE Edgeline servers have the memory, storage and compute power of data center class devices in a hardened package so they can be deployed in harsh environments. NVIDIA GPUs can be installed in HPE Edgeline servers, which enhances the ability of the server to perform complex calculations, which boosts the analytics capability that SAS ESP provides.

VIDEO ANALYTICS AT THE EDGE

As stated in the Introduction, the solution is made up of SAS Event Stream Processing, HPE Edgeline servers, and NVIDA GPU processors.

Specifically, the use case is one of video recognition and analytics processing based on what the server 'sees' through a camera that's attached to the server.

Figure 1 is graphic that depicts the logical flow of data. Information is observed using a video camera that's connected to an HPE Edgeline EL4000 server. The HPE Edgeline server is running SAS ESP, which recognizes a ball that's being controlled remotely by a user. SAS ESP then uses the NVIDIA T4 GPU to interpret where the ball is presently. SAS ESP then compares the ball's current location with the location from the last frame that was interpreted. Once it has done that, it can calculate the vector and velocity of the ball during the intervening time. SAS ESP then writes the ball's location, vector and velocity to an output file for further analysis later, if desired.

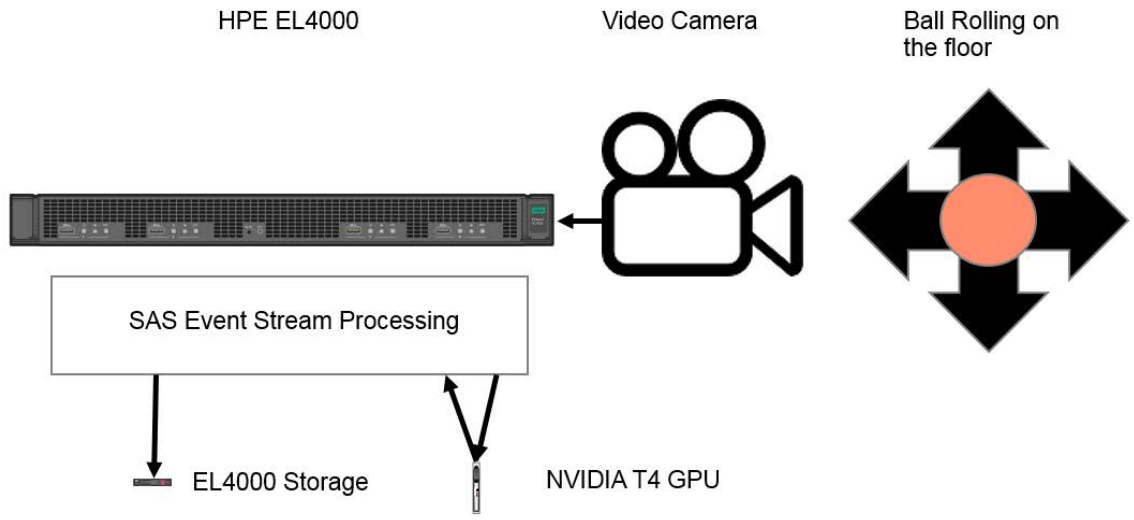

Figure 1. Logical flow of data from the video camera to the HPE Edgeline server running SAS Event Stream Processing

While this is a controlled use case, we can use this to represent a much broader range of industry relevant use cases. While, in this instance, ESP is trained to recognize the ball, it could likewise be trained to recognize more complex features. The specific scenario that this

Reference Architecture will highlight is an electricity utility use case, a company that generates and distributes electricity to customers.

In order to transmit the electricity that is generated at power plants, to its customers, an electrical utility needs miles of wires, both high tension as well as local area, lower voltage lines.

Today, utility companies often employ three service personnel working together simultaneously, looking for maintenance concerns such as cracks in the cables, rust on metal stanchions, rot on wooden poles and vegetation encroaching on the power lines. The inspection of power line infrastructure can be done in a limited way from the ground by visual inspection, but more typically it is done aerially, with a pilot flying a helicopter, another taking video or images of the power lines, stanchions and vegetation, and the third person in the control room to review the video feed looking for anomalies and providing feedback to the pilot and videographer.

In our smart edge scenario, we would replace these three people with a single person. That service person would pilot a drone with an attached camera for video capture. The drone would transmit the video feed back to an HPE Edgeline server located in the service vehicle supporting the drone pilot. The HPE Edgeline server, running SAS ESP, would interpret the video feed looking for cracks in the power lines, rust or rot on the power line stanchions and encroaching vegetation. Only when an issue is found, would any data need to be transmitted to the central location. That data would include latitude, longitude and type of anomaly. As the data was captured, if so desired, it could also be retained on storage SSD drives, which could be offloaded at the end of each shift, when the network available would be the facility's local area network.

Out of this one use case using SAS ESP to do video analytics, additional use cases become myriad.

In general, there are seven considerations why all data generated at the edge does not necessarily need to be transmitted to the data center (DC^2). They are:

1. Latency - The large amount of data can cause a delay between when the data was collected and when it arrives in the DC.
2. Bandwidth - The larger the amount of data to be transferred, the more bandwidth is required to transmit it in a given period of time. This problem is exacerbated if the data needs to travel outside of an enterprise's LAN.
3. Cost - Expanding bandwidth is expensive. This is especially true if a WAN is required.
4. Compliance - Certain types of companies have regulations they must follow when transmitting data. For instance, health care companies dealing with patient data must comply with HIPPAA regulations.
5. Security - Any time data is placed on a wire, it is subject to unauthorized access. This is especially true if the data is transmitted to remote locations.
6. Duplication of data - The data is duplicated on each server to which it's copied. This increases the overall storage space required.
7. Reliability - The farther data is carried from the edge, the more single points of failure are introduced. Performing as much processing as possible at the edge ensures reliability remains high.

In addition to our electric utility example, there are other industries for which video analytics would provide a return on investment and, for many, would buttress and

[2] Seven Reasons to Compute at the Edge: https://www.engineering.com/IOT/ArticleID/15540/Seven-Reasons-to-Compute-at-the-Edge.aspx

strengthen the analysis being performed. Each of those industries to which edge computing is applied has unique requirements and would deploy specialized clusters of computers, networking and storage devices to take advantage of those different demands.

A few examples of industries' video use cases that are driving the application of edge computing include, but certainly not limited to:

1. Theft and crime detection - Determine when people enter secure areas or commit acts of theft.
2. Customer insights - Intelligent systems can help stores increase business by providing a range of insights, such as time in store, number of customers, movement patterns and helping to determine the most popular areas and items.
3. Manufacturing - Increase the quality of manufactured parts. As an example, a high technology company that produces computer chips is using video analytics to ensure parts being manufactured will function when the process is complete. This company doesn't waste the resource finishing a part that will not be viable, instead they remove the part from production as soon as the anomaly is discovered. Hewlett Packard Enterprise also uses video analytics in the QA process when manufacturing HPE ProLiant servers.
4. Smart City use cases - Traffic lights and geographical scope traffic monitoring and flow control are just a few of the use cases for municipalities.
5. Public transportation safety - Minimizing the risks in public transportation hubs. Accident and incident detection.
6. Tourism - Can help cities increase and improve tourism. Ensuring the most popular tourist routes are free flowing increases the tourist's experience and encourages others to visit.

Each of the above industries can find advantages by using video analytics to process and analyze data at the edge, and transmitting only a summary or an exception subset of that data to a core data center or cloud computing storage environment, for action and/or later analysis.

SOLUTION OVERVIEW

SAS and Hewlett Packard Enterprise have partnered for over 32 years to help mutual customers solve their most difficult business problems. Both companies were early entrants of their respective IoT Partner ecosystems. They enjoy a close R&D relationship, and Hewlett Packard Enterprise engineering works closely with SAS to conduct rigorous tests, of which this paper is a product. Together, SAS and Hewlett Packard Enterprise have over 10,000 joint customers running decision support systems.

Both companies have an end-to-end product offering that allows the specific customer to determine the deployment model that will best meet their unique needs.

Both companies have technologies that are deployed from the edge to the core or cloud. The ability to deploy the right solution for a customer's individual requirements allows the customer to tailor a solution to their needs.

Hewlett Packard Enterprise has computer systems to fit in the right spot, from the HPE Edgeline Converged Edge systems, to data center systems that are designed from the ground up for robustness and resilience. These data center systems include HPE Synergy, HPE Apollo, HPE Superdome Flex and HPE ProLiant servers.

Hewlett Packard Enterprise also manufactures the networking solutions required to connect the entire hardware solution together, enabling a customer to set up a total end-to-end solution.

Like Hewlett Pacakrd Enterprise, SAS Institute has numerous products designed to address the needs of any type of customer. SAS Institute provides software applications, in addition to SAS Event Stream Processing (ESP) that includes business intelligence, data integration, fraud management, financial intelligence and IT management.

SAS EVENT STREAM PROCESSING

At the edge, we have SAS Event Stream Processing (ESP) for inference and alerting. SAS ESP can be used for everything from initial collection with a lighter level of analytics and alerting to deeper, more thorough analytics including, but not limited to, control of components on the factory floor, traffic lights and flow control in a Smart City, etc.

As mentioned earlier in this paper, SAS ESP can easily make use of the GPU capability that can be installed in the HPE Edgeline systems. All that is required to enable GPU functionality within SAS ESP is to include 2 or 3 keywords. Having SAS ESP make use of the system's CPU requires nothing more than removing those 2 to 3 keywords.

Additionally, if required, although not part of this test scenario, SAS offers SAS Visual Analytics® which can be deployed at the edge. Visual Analytics allows for visualization of the processes as they are executing at the edge, closer to the data collection points.

HPE EDGELINE CONVERGED SYSTEMS

At the edge, Hewlett Packard Enterprise offers HPE Edgeline Converged Edge systems to connect to, analyze, and store data from anything that can be connected to the digital world. The HPE edge family of systems spans from intelligent gateways that connect, convert, and perform simple analyses to true converged edge systems that can provide data center-like compute, connectivity and remote systems management. These systems are architected to address increasing performance requirements, allowing a customer to choose the specific model to address the unique requirements at each and every data collection point in the process, ultimately providing the ability to run data center apps at the edge.

HPE Edgeline Converged Edge systems have the resources required to allow for larger memory, robust storage and networking options, and video processing capabilities to be deployed right at the edge, where they can be most useful. With video processing, we need to process a minimum number of frames per second in order to be effective. While some AI workloads can be handled using a computer's CPU resources, our testing indicates that this would result in a sub-optimal configuration resulting in modification to the operational design, or worse, having to transmit all video data to the data center or cloud. Transmitting data in this manner will increase the cost associated with arranging for a higher bandwidth network. It further has the undesired effect of increasing the latency associated with analyzing the data. In the manufacturing environment discussed above, it might make the utilization of video analytics untenable.

Our testing (results are below in the Workload Results section) has shown that providing the ability to install an NVIDIA Tesla T4 processing unit addresses the need for high speed compute for video processing at the edge.

SOLUTION COMPONENTS

SOFTWARE AT THE EDGE

The following, table 1, is a list of software used in our environment[3].

Table 1. Software

[3] Information courtesy of NVIDIA, found at this URL: https://www.nvidia.com/en-us/data-center/tesla-t4/

Software	Version
SAS Event Stream Processing	6.1
SAS Viya	3.3
Red Hat Enterprise Linux	7.5
NVIDIA-SMI	410.104
NVIDIA Driver Version	410.104
NVIDIA CUDA Version	10.0

HARDWARE AT THE EDGE

HPE Edgeline EL4000 Converged Edge System

At the edge, we utilized an HPE Edgeline EL4000 Converged Edge system. The HPE Edgeline EL4000 can be configured with 1 to 4 server cartridges. Each server cartridge runs an industry standard operating system, such as Red Hat Enterprise Linux® or Microsoft Windows Server®.

HPE ProLiant m510 server cartridge

The HPE ProLiant m510 server cartridge is available with either an eight core or a sixteen core Intel Xeon processor and up to 128GB of memory. When paired with the available NVIDIA Tesla T4 GPU, this server cartridge can perform inference operations at the edge, allowing rapid enhancement of the models being executed for capture, control and alerting. SAS Event Stream Processing, SAS Viya and SAS Visual Analytics have been validated to work on the HPE ProLiant m510 Server cartridge.

NVIDIA Tesla T4 GPU

The HPE EL4000 Edgeline Converged Edge system contains 4 PCIe slots, one associated to each server cartridge. Because we conducted tests using two (2) HPE ProLiant m510 cartridges, we inserted two (2) NVIDIA Tesla T4 GPUs.

Table 2 contains the characteristics of the NVIDA Tesla T4 GPU.

Table 2. NVIDIA Tesla T4 GPU features

Metric Type	Metric Name	Metric
Performance	Turing Tensor Cores	320
	NVIDIA CUDA Cores	2,560
	Single Precision Performance (FP32)	8.1 TFLOPS
	Mixed Precision (FP16/FP32)	65 FP16 TFLOPS
	Int8 Precision	130 Int8 TOPs
	Int4 Precision	260 Int4 TOPs
Interconnect	Gen3	X16 PCIe
Memory	Capacity	16GB GDDR6
	Bandwidth	320+ GB/sec
Power	Maximum Power Consumption	70 watts

WORKLOAD DESCRIPTION

As stated earlier in this paper, the workload was a series of screen captures that were generated as a video camera watched a ball rolling around on the floor. The screen captures were recorded 30 times per second. The ball's direction and speed were deduced by understanding what location the ball was located in during the prior screen capture and where it is for the present screen capture. As each screen capture was processed to determine where the ball was on the floor, SAS ESP then calculated the center of the ball on the X and Y coordinates. Once the ball's present location was calculated, SAS ESP then calculated how far and in what direction it had rolled since the last X and Y coordinates.

Figure 2 is a depiction of the data flow within SAS ESP during the test scenario. This depiction was captured using SAS Event Stream Processing Studio.

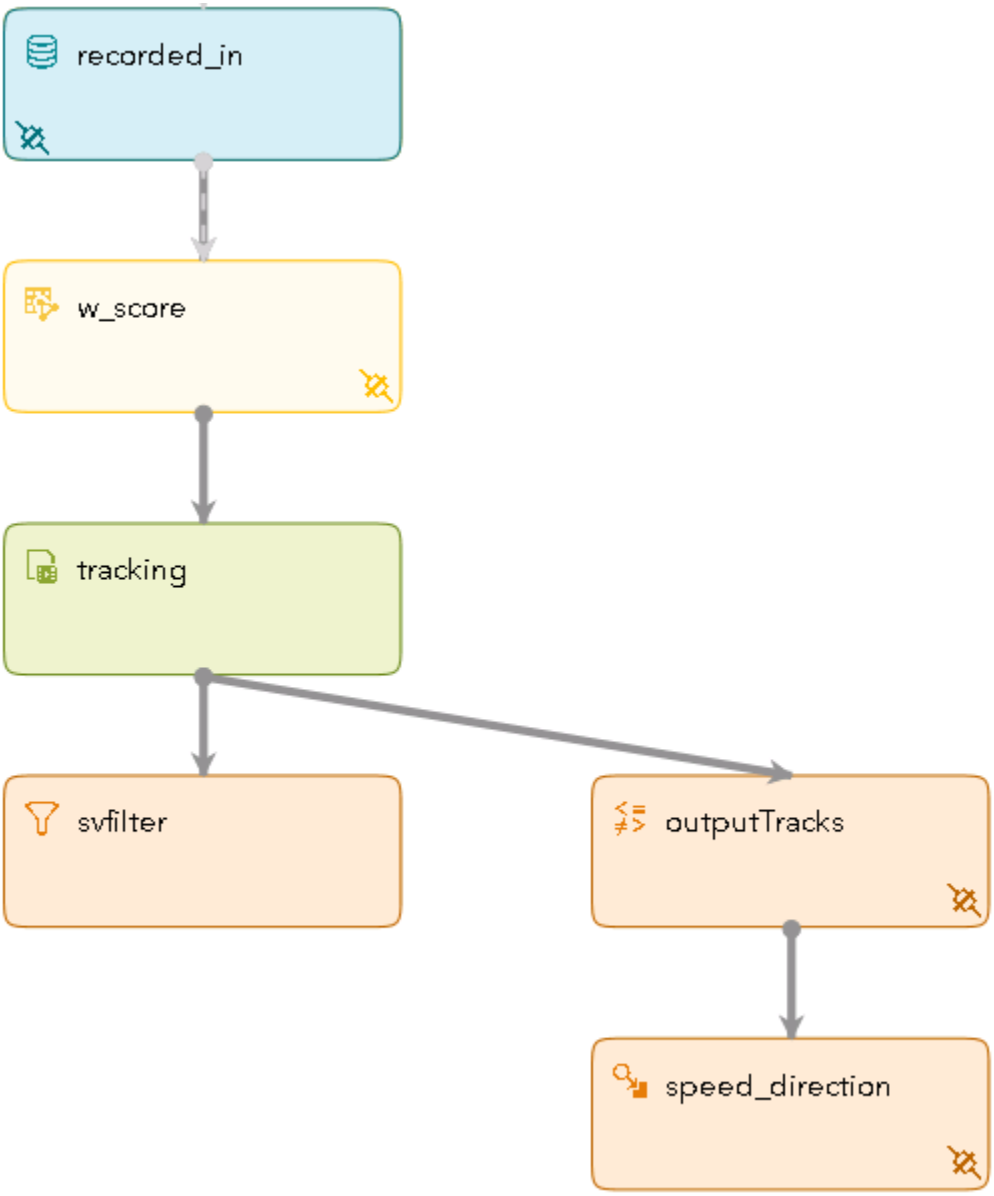

Figure 2: SAS Event Stream Processing job flow

Each of the rounded rectangles contained within the data flow represents a task that SAS ESP performs.

The first rounded rectangle, named recorded_in, is the task that reads the input file, orders the data contained within the input file and passes it along to w_score. The rapidity with which we read the data input file can be varied. In the original test, we used a value of 30 to tell it to query the video camera 30 times every second. Since, during this testing, we're reading a data file that was captured during the original run, we can vary the speed with virtually unlimited values. We used the same input file containing the same data for all tests to ensure consistency between tests, thereby ensuring that SAS ESP was performing the same tasks for each test, which guarantees uniformity between tests. Uniformity between tests allows direct comparisons to be made.

Input to recorded_in is the name of the file to be read, the rate at which to read and ingest records from the input file and the number of times to read the input file.

After data is read and ingested to SAS ESP, recorded_in passes that data to w_score. The sub-process w_score simply identifies the fields that were read by recorded_in, orders them and passes the data along to the sub-process tracking.

The sub-process tracking task is to find the coordinates of the ball at present and pass them along to svfilter and outputTracks.

The sub-process svfilter's only task is to determine which screen shots were to be processed. This was set to 100%, but in other scenarios it could sample a lesser amount. For instance, if it was determined that 30 frames per second was too many frames and that the intervening frames probably would not be useful, this value could be set to 66% to enable 2 out of every 3 frames to be examined. Or it could be set to 50% so that only half of the frames would be examined.

The sub-process outputTracks keeps the last position of the ball in memory, and calculates the current position of the ball on the floor and passes that along to speed_direction.

The sub-process speed_direction then takes those values and calculates the speed and direction of the ball along the floor. It then outputs a record to its log file, so that it can be reread later in case additional analysis needs to be completed after the fact.

While the original scan rate of the video camera was 30 frames per second, we wanted to determine the maximum number of frames that could be processed. As a result, we set this to 90 frames per second. What happens is that SAS ESP will self regulate. If it can't keep up with 90 frames per second, it will simply analyze fewer frames per second and report that to its log file.

We then ran the test with 1, 2, 3 and 9 concurrent SAS ESP processes. These processes were all reading the same input file, but the reads are non-blocking. We ran simultaneous ESP instances because we wanted to ensure that we were maximizing the available throughput.

We also wanted to demonstrate the differences in performance when we ran with an HPE ProLiant m510 cartridge that had more cores, albeit with a lower clock speed versus one that had fewer cores, but a faster clock. The tests were run on one (1) HPE ProLiant m510 cartridge that had 8 cores running with a clock speed of 2.0GHz and another HPE ProLiant m510 cartridge that had 16 cores running with a clock speed of 1.7GHz.

WORKLOAD RESULTS

The first graph, figure 3, shows the performance comparison with the number of frames per second able to be processed on the 8-core, 2.0GHz system. This is the number of frames per second across the entire system. As you can see the total number of frames the system can process increases as the number of concurrent streams is increased. This is expected behavior, and demonstrates how the difference increases as the number of concurrent streams increases.

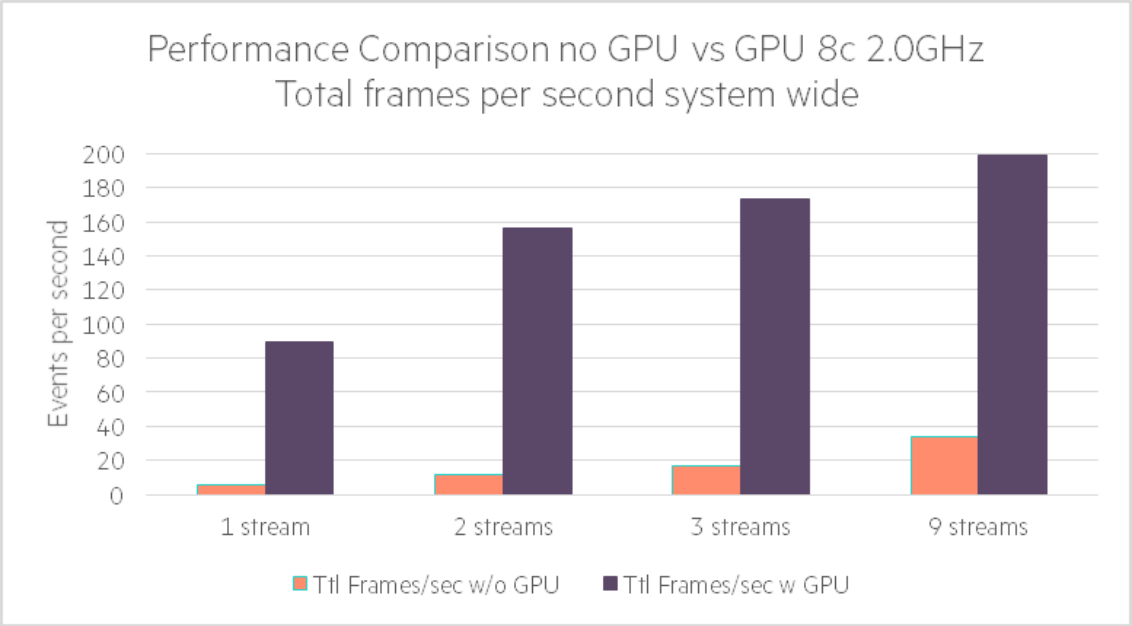

Figure 3. Performance comparison between GPU thoughput and CPU throughput in number of frames per second processed for the 8-core 2.0GHz server

There are a number of interesting things to note in the above graph. First, when determining the number of frames per second for a single stream, the GPU delivered 15 times the performance able to be delivered using the system's CPU only. We were able to analyze 90 frames per second using the GPU versus only 6 frames per second using the system's CPU.

Next, when looking at the total number of frames per second, our focus is drawn to the 9 concurrent SAS ESP instances. Here, when we compare the GPU's performance versus that of the CPU, we find that the GPU was able to out perform the CPU by a factor of 5.85.

The next graph, figure 4, is the same type of graph, only this time we show the per-process, average number of frames per second. As we scale from 1 to 2 concurrent ESP processes, the number of frames being processed on a per-process basis will be reduced, which is to be expected, because of latency introduced based on system resources being consumed.

We were able to remain well above the 30 frames per second with all of the runs using the GPU with the exception of the 9 concurrent SAS ESP instances. In fact, even with 9 simultaneous SAS ESP instances we were able to deliver 22 frames per second, per process using the GPU. We were able to get a maximum of 4 frames per second, per process when using the CPU.

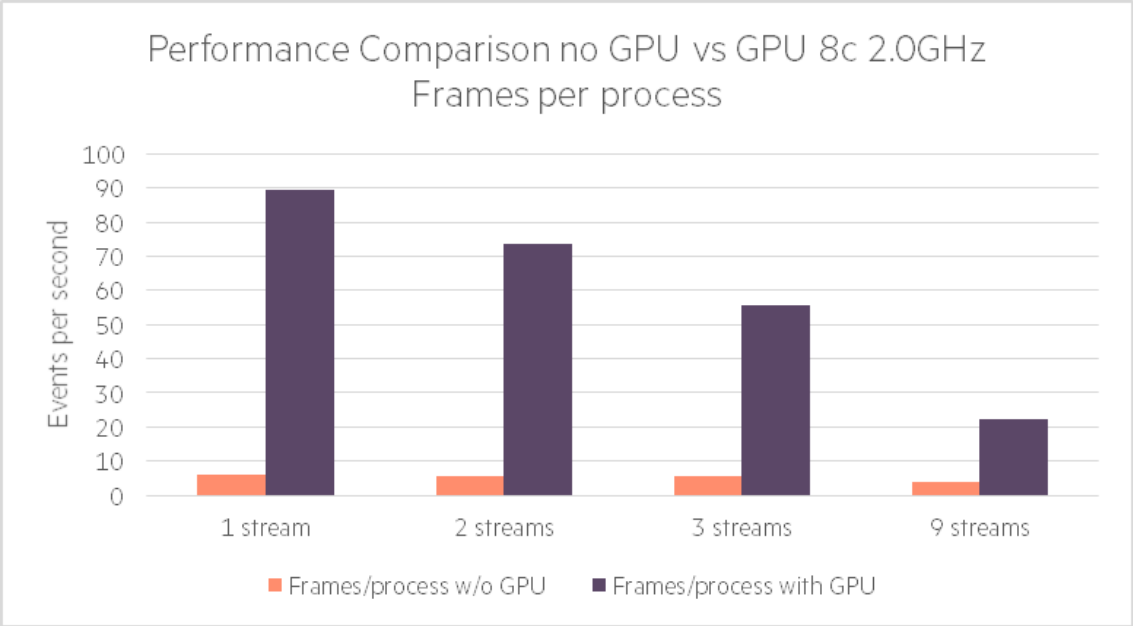

Figure 4. Performance comparison between GPU thoughput and CPU throughput in number of frames per second, per process on the 8-core 2.0GHz server

The next graph, figure 5, is the same type of graph, however, this time we are running the tests on the 16-core 1.7GHz system. This graph shows the total number of frames per second that the system was able to achieve.

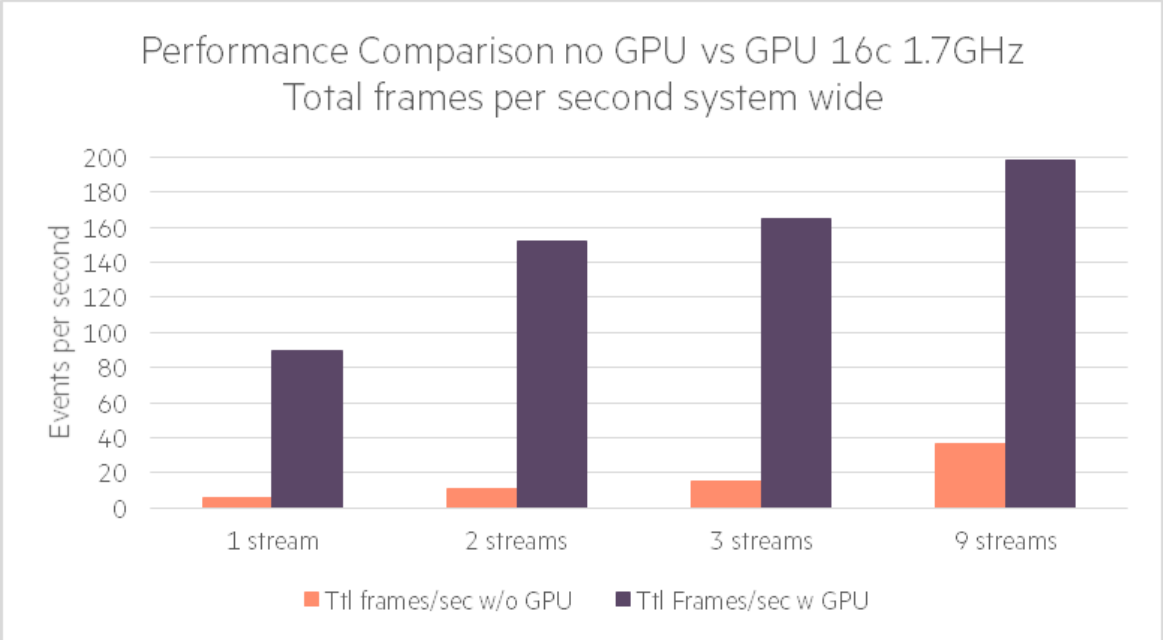

Figure 5. Performance comparison between GPU thoughput and CPU throughput in number of frames per second processed for the 16-core 1.7GHz server

The reduced clock speed did have an impact when using the CPU. When using the CPU, the number of frames per second was reduced to 5 per second. This was expected behavior. However, the number of frames per second remained the same when using the GPU to perform the calculations on the system with the reduced clock speed. The difference between the CPU and GPU performance was increased to 18x when running a single stream.

The next graph, figure 6, is the number of frames per second, per process on the 16-core 1.7GHz server.

One of the interesting things to point out is that having more cores, even at a reduced clock speed helped us when we ran additional simultaneous SAS ESP instances, which used more of the cores. When we get to the 9 SAS ESP instances, the GPU performance was still 5.53x that of the CPU only performance. We attribute that to the reduced contention for CPU resources brought on by additional cores with which to process concurrent SAS ESP processes.

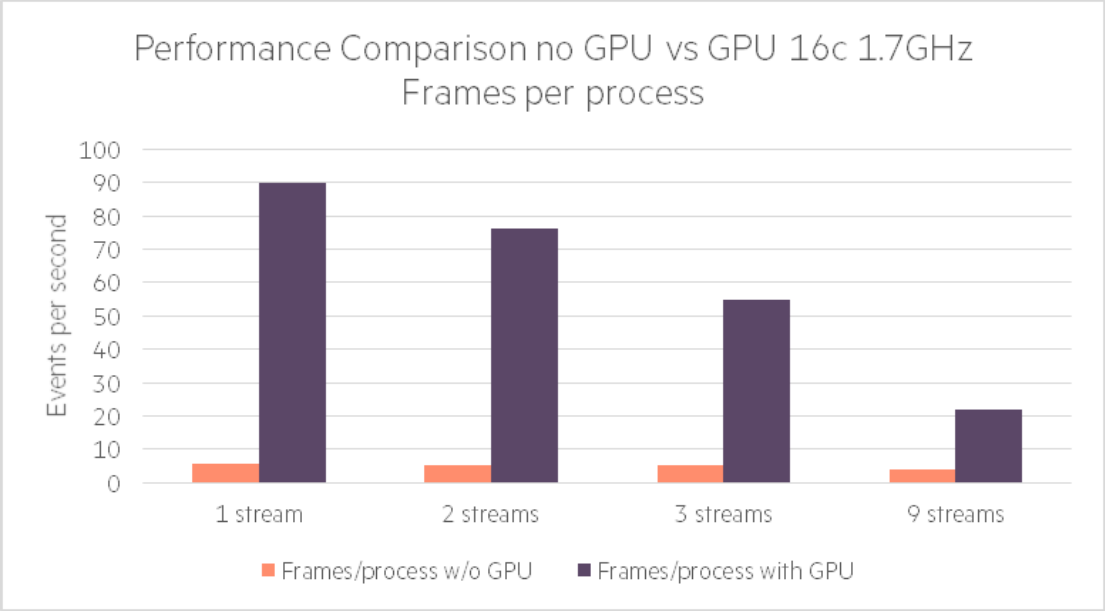

Figure 6. Performance comparison between GPU thoughput and CPU throughput in number of frames per second, per process on the 16-core 1.7GHz server

The next question we asked was, what impact does the ability to process these frames per second at these rates have on the total run time required to retire the input file. The following, figure 7 is the same graph as above for the 8-core 2.0GHz system, only we have now overlaid the time to complete the tests on the graph.

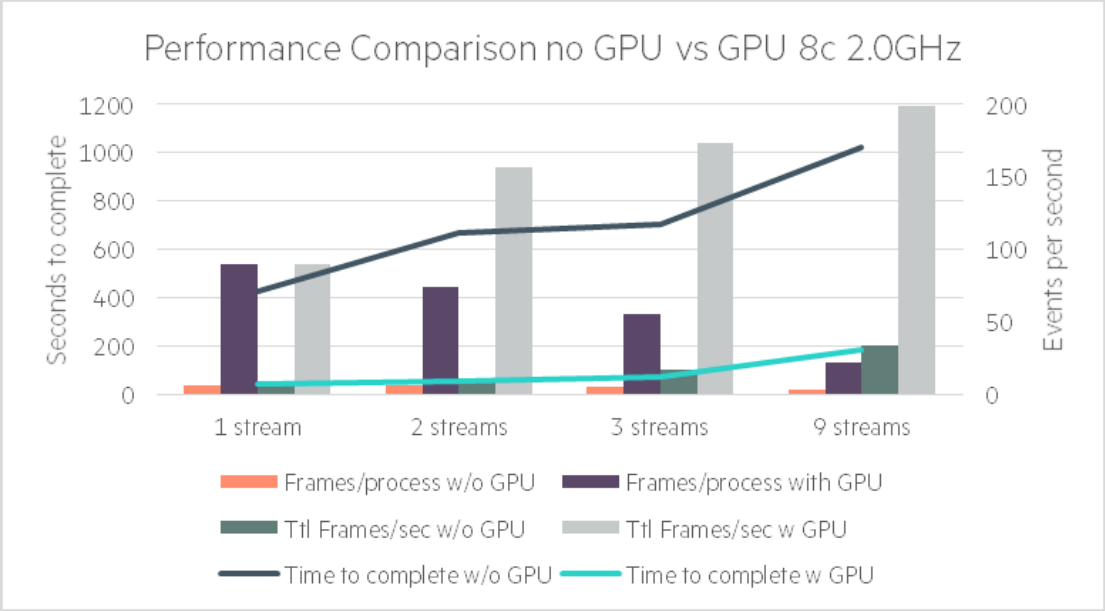

Figure 7. Performance comparison with time to complete the test overlaid while running on the 8-core, 2.0GHz system

As you can see, the real impact is the time to complete the tests. This translates quite linearly, into the inability of the CPU to be able to process frames at a rate fast enough to reduce or eliminate the need to transfer the frames to the data center or cloud for analysis. Here we have a comparison time between when we used a GPU versus when we used the system's CPU for the calculation processing. When running all 9 concurrent SAS ESP processes the GPU run time was 5.5 times less than when using the system's CPU. When we ran a single SAS ESP session, the difference is more stark with the difference being the GPU time was 9.28 times faster than the time when using the system CPU. The largest time difference occurred when we were running 2 SAS ESP instances concurrently. That time factor had the GPU result completing 11.7 times faster than the CPU time to complete.

The following, figure 8 shows the time to complete the tests on the 16-core, 1.7GHz system.

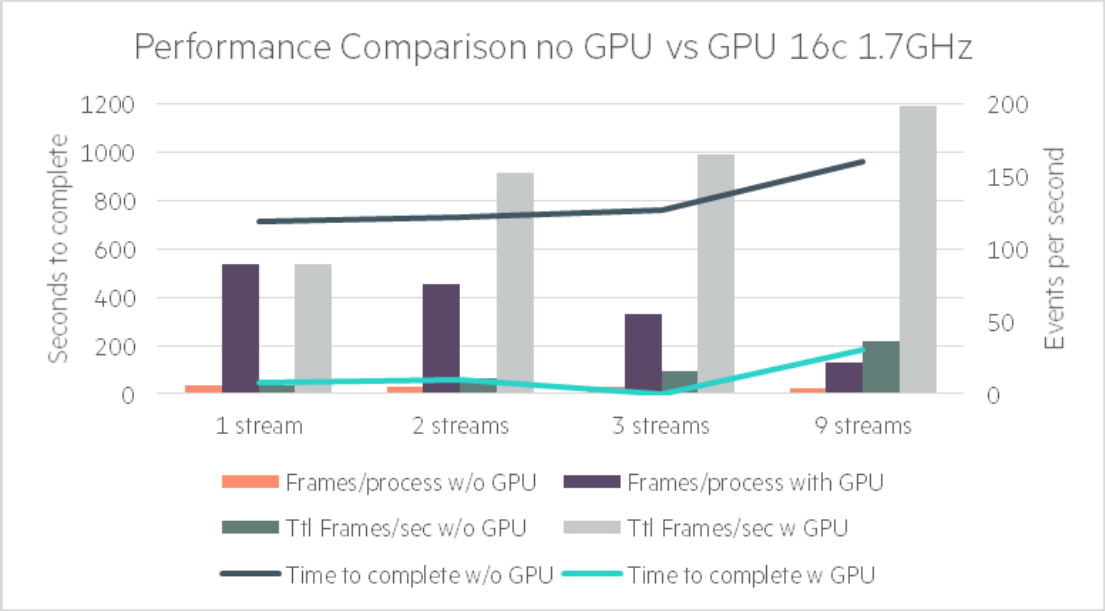

Figure 8. Performance comparison with time to complete the test overlaid while running on the 8-core, 2.0GHz system

NVIDIA provides software that can be used to query the performance of the installed GPU. There are quite a few metrics from the GPU that can be captured. Those metrics include, but are not limited to power consumed in watts, temperature of the GPU in degrees Celsius, percentage of CPU consumed within the GPU, the percentage of GPU memory being consumed, the clock rate of the GPU and the clock rate of the memory.

The two main metrics that tend to throttle the GPU are power consumed, which has a threshold of 70 watts, and temperature, which has a threshold of 80 degrees Celsius nominal. When either of those conditions are encountered, the NVIDIA hardware will start to throttle. The prevalant throttling mechanism is to reduce memory speed. The secondary method is to throttle the amount of power being consumed.

The following graph, figure 9 shows the GPU's performance when coupled with the 8-core 2.0GHz system. On the left axis is the percent GPU busy, both peak and average across the entire test. Peak busy only needs to occur once during a test in order for it to be recorded. Average busy takes into account the amount of time that the GPU was busy over the entire duration of the test. This includes ramp up and ramp down, and as a result may be skewed lower than one would expect.

On the right axis is the peak watts being consumed along with the peak temperature in degrees Celsius.

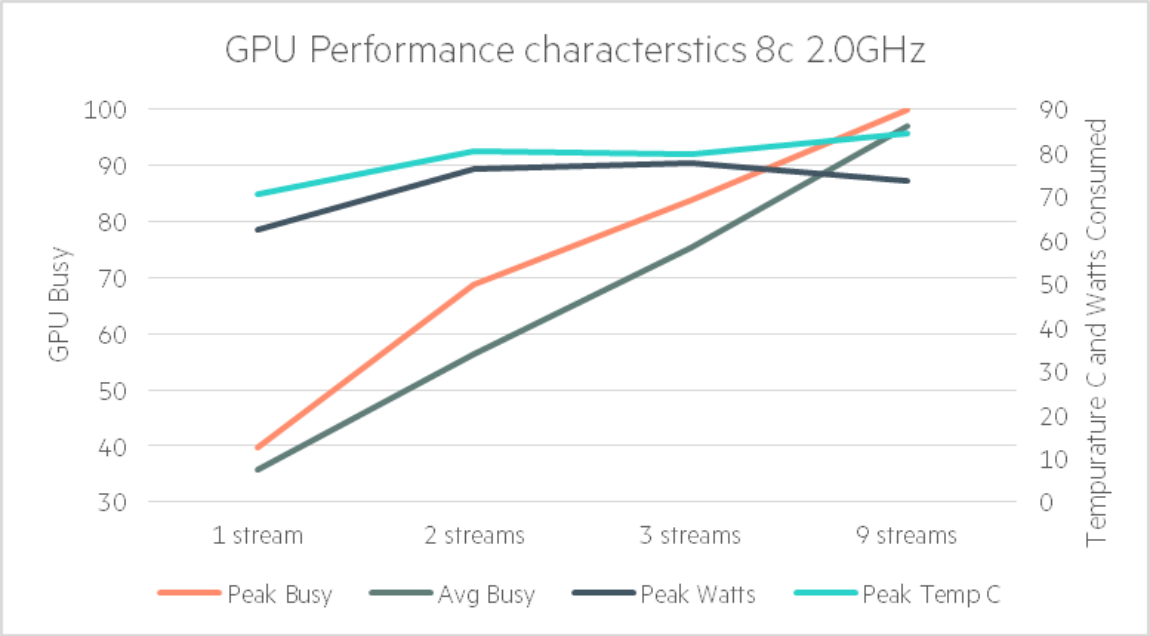

Figure 9. GPU metrics collected during the tests when the GPU was utilized on the 8-core 2.0GHz system

The following graph, figure 10, is the same type of graph, with the only difference being that this was collected on the system with 16-cores running at a 1.7GHz clock speed.

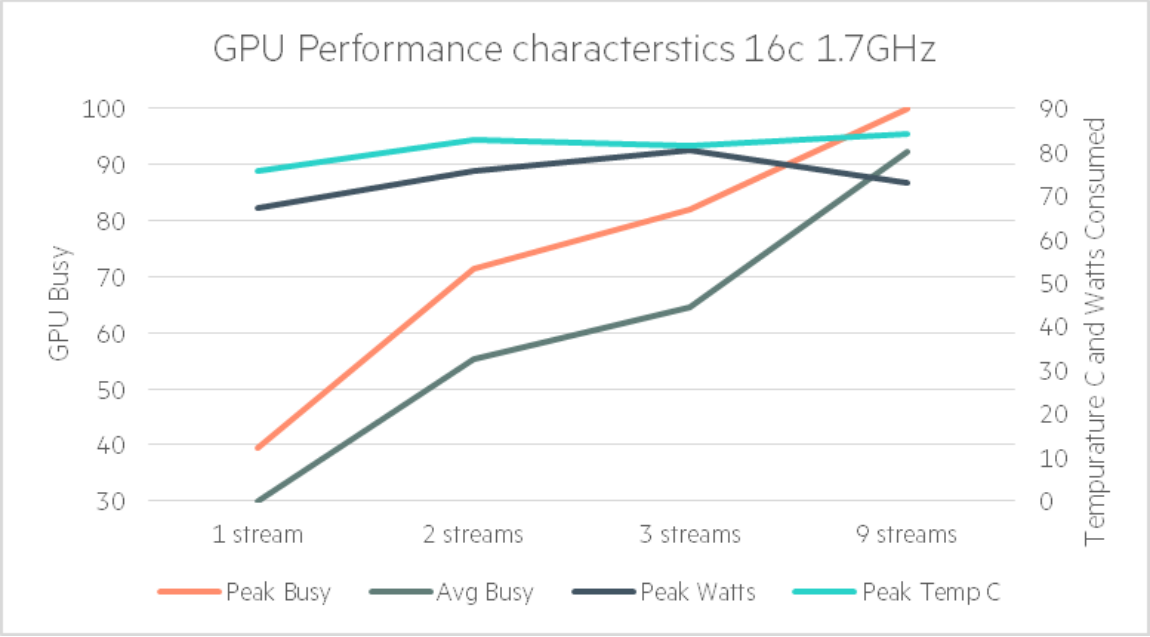

Figure 10. GPU metrics collected during the tests when the GPU was utilized on the 8-core 2.0GHz system

ANALYSIS AND RECOMMENDATIONS

Our analysis has shown the ability of an HPE Edgeline Converged Edge System, when coupled with a NVIDIA Tesla T4 GPU and SAS Event Stream Processing, to deliver video analytics at the edge, where the video is being collected. Processing and analyzing video at the edge has the potential to eliminate or at least reduce the amount of data that needs to be transmitted from the edge to the data center or cloud. Reducing or eliminating the need to transmit data from the edge to the data center or cloud also has the advantage of eliminating the latency introduced by the transmission of the data. The reduction of latency and additional speed in processing video analytics at the edge creates far more opportunities to make video processing viable at the edge. This additional processing capability is provided in a hardened compute server that can be deployed anywhere from factory floor, electrical vault, service person's truck or stadium venue.

For scaling purposes, a customer should to decide how many frames per second will yield the results. As shown in the above performance graphs, a single SAS ESP process running on an HPE ProLiant m510 cartridge is able to analyze 90 frames per second. If a reduced number of frames is required, more SAS ESP processes can be run concurrently. Even when we were running 9 simultaneous SAS ESP processes, we were able to analyze 22 frames per second.

If a customer wishes to deploy fewer resources at the edge, they may choose the HPE EL1000 Edgeline Converged Edge System with a single compute cartridge. And since the EL1000 utilizes the same family of cartridges as the HPE Edgeline EL4000, they can tailor the solution for their specific requirements. Further, as with the HPE Edgeline EL4000, the HPE Edgeline EL1000 can be fitted with an NVIDIA Tesla T4 GPU for enhanced video analytics right at the edge.

While we utilized 2 of the available 4 cartridge slots during our testing, one at a time, the second HPE ProLiant m510 cartridge could be run concurrently with the one that was tested. If further compute resources are required an additional two (2) cartridges, along with NVIDIA Tesla T4 GPUs, can be placed in the HPE Edgeline EL4000 frame.

If even more processing power is required at the edge, Hewlett Packard Enterprise offers our HPE EL8000 Converged Edge system. This is a blades-based system and offers the Intel Gold 6212U processor with 24 cores running at 2.4GHz. The server blades offered in the HPE Edgeline EL8000 come in two different configurations. Those are 1U and 2U server blades. The HPE Edgeline EL8000 can house either four (4) 1U server blades or two (2) 2U server blades. The 1U server blades can accept a single NVIDIA Tesla T4 GPU, while the 2U server blade can accept up to four (4) NVIDIA Tesla T4 computational accelerators, making this a video analytics workhorse.

Scaling SAS' Event Stream Processing software at the edge is as easy as deploying a second, third, fourth, etc. instance of Event Stream Processing on the same cartridge or server blade. This ability of deploying additional copies of SAS Event Stream Processing within a single operating environment will work extremely well as we scale the number of GPUs attached to the HPE Edgeline EL8000 server blades. Additionally SAS Event Stream Processing can be placed on any number of servers within a customer's environment.

SUMMARY

At the edge, Hewlett Packard Enterprise has a large portfolio of converged edge systems that will accept the NVIDIA Tesla T4 computational accelerator. This allows the selection of hardware to fit each unique requirement. The product portfolio starts with cartridge systems and progresses up to four (4) server blades in a single chassis with four (4) NVIDIA Tesla T4 GPUs in the chassis or two (2) 2U server blades, with each server blade able to be configured with up to four (4) NVIDIA Tesla T4 GPUs.

This breadth of offering allows a customer to acquire a server that fits both budget and workload profile.

When combined with SAS Event Stream Processing at the edge, the total offering provides the ability to capture and react to massive amounts of video processing, automating tasks that were heretofore either exceedingly expensive or untenable.

CONTACT INFORMATION

Your comments and questions are valued and encouraged. Contact the author at:

Bob Augustine
Hewlett Packard Enterprise
512.319.0167
bob.augustine@hpe.com

Sri Raghavan
Hewlett Packard Enterprise
919.914.0069
raghavan@hpe.com